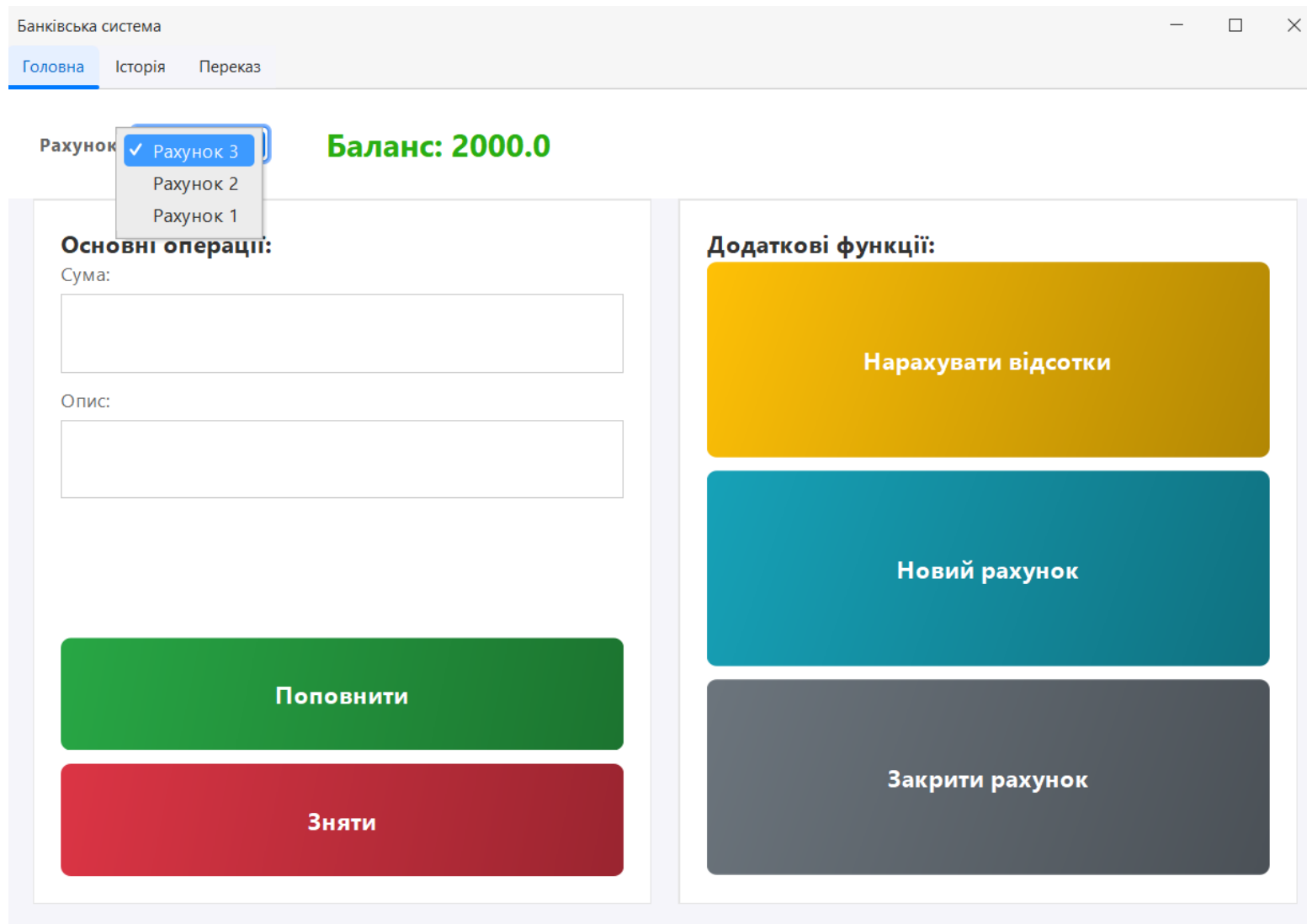


Банківська система на Kotlin зі Swing-інтерфейсом

Демонстрація процесів: створення рахунків, транзакції, потокобезпечні операції, FlatLaf-інтерфейс



Основні операції:

Сума:

100

Опис:

Поповнення

Успіх

Поповнення успішне!

ОК

Успіх

Зняття успішне!

ОК

Успіх

Рахунок успішно закрито!

ОК

Успіх

Переказ успішний!


ОК

Повідомлення про успішні операції в додатку: поповнення рахунку, зняття коштів, переказ на інший рахунок, закриття рахунку


Інтерфейс додатку (випадаючий список для перемикавання між рахунками), баланс, основні операції, додаткові функції та вкладки Головна, Історія, Переказ

Реалізовано основні функції банківської системи


Керування рахунками, операції та детальна історія транзакцій




Управління рахунками:
створення з власником і початковим балансом, закриття, блокування




Фінансові операції:
поповнення, зняття, внутрішні перекази між рахунками



Нарахування відсотків:
місячні відсотки на всі активні рахунки



Історія транзакцій:
таблиця з датою, типом, сумою, рахунками відправника/отримувача, описом, балансом, статусом



Збереження транзакцій: в пам'яті з обмеженням у 1000 записів для продуктивності

Банківська система

Головна Історія Переказ

Дата	Тип	Сума	З рахунку	На рахунок	Опис	Баланс	Статус
2025-12-02 12:30:48	Поповнення	200,00	Зовнішнє джерело	Цей рахунок	Депозит: Початков...	1200,00	Успіх
2025-12-02 12:30:48	Зняття	50,00	Цей рахунок	Користувач	Зняття: Обід	1150,00	Успіх
2025-12-02 12:30:48	Зняття	100,00	Цей рахунок	Користувач	Переказ на Рахуно...	1050,00	Успіх

Оновити історію

Відображення історії руху коштів по рахунку із зазначенням дати та часу операції, типу, суми, походження коштів, опису, балансу після виконання та статусу виконання операції

Користувацький інтерфейс та управління рахунками

Огляд GUI-компонентів, кастомних елементів та потоків управління рахунками



Головне вікно з вкладками
Головна, Історія, Переказ



FlatLaf (FlatMacLightLaf) для
сучасного Мас-подібного
вигляду



Таблиця Історія: список
транзакцій з можливістю
оновлення



Форма Переказ:
міжрахункові платежі із
полями отримувача та суми



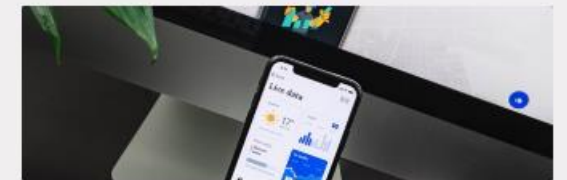
Кастомні елементи:
GradientButton, панелі з
білим фоном і рамками



Управління рахунками
через комбобоксы для
поточного та рахунку
отримувача



Діалог для створення нових
рахунків



Підсумок: інтуїтивний
інтерфейс та кастомні
контролі для швидкого
управління рахунками

Логіка операцій та потокобезпечність

Коротко про валідацію операцій, запис транзакцій і механізми безпеки при багатопоточності

The screenshot shows a web application window titled "Банківська система". It has three tabs: "Головна", "Історія", and "Переказ", with "Переказ" being the active tab. The form contains the following fields:

- З рахунку:** A dropdown menu with "Рахунок 1" selected.
- На рахунок:** A dropdown menu with "Рахунок 3" selected.
- Сума:** A text input field containing "500".
- Опис:** A text input field containing "Повернення боргу".

At the bottom of the form is a large purple button labeled "Виконати переказ".

- 1 Операції:** депозит, зняття, переказ — автоматичне додавання запису до історії транзакцій для кожної дії
- 2 Перевірки перед виконанням:** доступність коштів та статус рахунку (відкритий чи закритий)
- 3 Нарахування відсотків:** виконується одночасно для всіх рахунків — централізований цикл обробки
- 4 Потокобезпечність:** використання **synchronized** для атомарних блоків та **ConcurrentHashMap** для спільного сховища
- 5 Гарантії:** відсутність конфліктів і коректність даних при паралельній роботі завдяки комбінованим примітивам синхронізації
- 6 Підсумок:** валідація → логування → атомарність через **synchronized** + **ConcurrentHashMap** для масштабованості

Вкладка для переказу коштів між рахунками: випадуючі списки для вибору рахунків, текстові поля для опису та суми, кнопка Виконати переказ

Переваги та технічні деталі реалізації

Коротко про бізнес-логіку, масштабованість та UX для розробників

Поетапне створення нового рахунку:

Input

Введіть ID нового рахунку:

Тестовий

OK Cancel

Input

Введіть початковий баланс:

5000

OK Cancel

Input

Введіть ім'я власника:

Владислав Бакало

OK Cancel

Успіх

Рахунок успішно створено!

OK

Банківська система

Головна

Рахунок 3

Рахунок 2

Рахунок 1

✓ Тестовий

Баланс: 5000.0

 Комплексний банківський процес з зрозумілим і сучасним інтерфейсом	 Реальна бізнес-логіка: моделі BankAccount, Transaction, Bank-менеджер
 Масштабованість: легко додати функції або інтегрувати БД	 Підготовка до безпечної багатопотоковості для стабільної роботи
 Технології: Kotlin , GUI — Swing + FlatLaf	 UX: градієнтні кнопки та миттєве оновлення інтерфейсу

Мета додатку

- Демонстрація повноцінної банківської системи з GUI.
- Можливість працювати з рахунками, переглядати історію транзакцій, робити перекази та нараховувати відсотки.
- Використання потокобезпечних операцій та сучасного оформлення інтерфейсу.